

CLAIMS

1 1. In an information processing system comprising a plurality of stacks each
2 comprising at least one stack frame, a method for detecting phases in a computer
3 program running at least one thread, the method comprising the steps of:

- 4 a) allocating space in memory for an activation count for each frame;
5 b) zeroing the activation count wherever the program creates a new stack
6 frame;
7 c) analyzing the stack for each thread and incrementing the activation count for
8 each frame; and
9 d) associating a phase with an activation whose activation count is non-zero.

1 2. The method of claim 1 wherein step c) further comprises logging activation
2 counts during each interval.

1 3. The method of claim 1 wherein the activation count is implemented by
2 reserving storage in each stack frame.

1 4. The method of claim 1, further comprising performing step performing step c)
2 at periodic intervals of time according to a system clock.

1 5. The method of claim 1 further comprising examining each stack to determine
2 the number of frames used in the stack and incrementing the activation count by the
3 number of frames.

1 6. The method of claim 1 further comprising ensuring that when a phase ends, an
2 action is performed immediately.

1 7. The method of claim 6 further comprising changing the return address to force
2 the program to call a designated procedure when the frame returns.

1 8. The method of claim 1 further comprising scheduling garbage collection after
2 each associated phase.

1 9. The method of claim 1 further comprising scheduling thread switches at phase
2 boundaries.

1 10. The method of claim 1 further comprising scheduling checkpoint operations
2 after each associated phase.

1 11. The method of claim 1 further comprising presenting a visualization of
2 program phase behavior.

1 12. The method of claim 1 further comprising resetting profile data at program
2 phase transitions.

1 13. The method of claim 1 wherein the activation count is represented by a single
2 bit, representing the presence or absence of a running phase.

1 14. The method of claim 1 further comprising implementing activation counts in a
2 side data structure.

1 15. The method of claim 1 wherein the activation count is implemented as an array
2 paralleling the stack.

1 16. A system for detecting phases in running computer programs, wherein the
2 program supports garbage collection, the system comprising:

3 a plurality of stacks each comprising at least one stack frame comprising an
4 activation counter; and

5 a processor comprising logic for:

6 zeroing the activation count wherever the program creates a new stack frame
7 and after garbage collection is performed;

8 analyzing the stack for each thread and incrementing the activation count for
9 each frame; and

10 associating a phase with an activation whose activation count is non-zero.

1 17. The system of claim 16 wherein the processor further comprises logic for
2 logging activation counts during each interval.

1 18. The system of claim 16 wherein the activation count is implemented by
2 reserving storage in each stack frame.

1 19. The system of claim 16 wherein the processor further comprises logic for
2 analyzing the stack for each thread and incrementing the activation count for each
3 frame at periodic intervals of time according to a system clock.

1 20. The system of claim 16 wherein the processor comprises logic for causing the
2 system to call a designated procedure when the frame returns.

- 1 21. The system of claim 16 wherein the memory comprises instructions for
- 2 causing the system to call a designated procedure when the frame returns.

- 1 22. The system of claim 16 wherein the processor comprises logic for examining
- 2 each stack to determine the number of frames used in the stack and incrementing the
- 3 activation count by the number of frames.

- 1 23. A computer readable medium comprising program instructions for:
 - 2 a) associating an activation count with each frame;
 - 3 b) zeroing the activation count wherever the program creates a new stack
 - 4 frame and after garbage collection is performed;
 - 5 c) analyzing the stack for each thread and incrementing the activation count for
 - 6 each frame; and
 - 7 d) associating a phase with an activation whose activation count is non-zero.

- 1 24. The computer readable medium of claim 23 further comprising program
- 2 instructions for logging activation counts during each interval.

- 1 25. The computer readable medium of claim 23 further comprising program
- 2 instructions for reserving storage in each stack frame for the activation count.

- 1 26. The computer readable medium of claim 23 further comprising program
- 2 instructions for examining each stack to determine the number of frames used in the
- 3 stack and incrementing the activation count by the number of frames.

1 27. The computer readable medium of claim 23 further comprising program
2 instructions for ensuring that when a phase ends, some action is performed
3 immediately.

1 28. The computer readable medium of claim 23 further comprising program
2 instructions for changing the return address to force the program to call a designated
3 procedure when the frame returns.

1 29. The computer readable medium of claim 23 further comprising program
2 instructions for scheduling garbage collection after each associated phase.

1 30. The computer readable medium of claim 23 further comprising program
2 instructions for scheduling thread switches at phase boundaries.